Pandemonium Illusion

Just another WordPress.com weblog

~~Amazon Associates~~
~~Google Adsense~~
~~Gravatar~~
~~SkimLinks~~
~~Wordpress Stats~~

← Ignore .pyc files in Subversion

Starting a new Django Project with Versioning →

Compress a PDF with pdftk

Posted on May 7, 2008 | 36 Comments

UPDATED (9/7/11): The package is not pdftk, but Ghostscript, and it's easily installable through homebrew on a mac.

Ok, so I don't know why this works, but it does and I haven't seen any decrease in quality. I suspect converting from PDF to PS and back just optimizes the PDF.

Requirements

- pdftk for linux

The Process

1. Convert your PDF to PS (this creates a large file
2. Convert the new PS back to a PDF

```
pdf2ps large.pdf very_large.ps
ps2pdf very_large.ps small.pdf
```

Results
large.pdf : 6.3MB
very_large.ps : 53.4MB
small.pdf : 2.4MB
Looks like pretty good compression to me.

If anyone knows why this works so well, please let me know.

Loading...

This entry was posted in Uncategorized. Bookmark the permalink.

36 RESPONSES TO "COMPRESS A PDF WITH PDFTK"

Jasper | July 16, 2008 at 9:19 pm | Reply

I guess there is some quality loss during the process. Losing lines and curves just having bitmaps …

dude | July 30, 2008 at 11:21 pm | Reply

pdftk is also able to uncompress and compress a pdf (see those

SEARCH IT!

Search

RECENT ENTRIES

- Homebrew and MySQL server
- Better way to use Django's SelectMultiple in Google App Engine for a ListProperty
- Django ImageField and FileField dynamic upload path in Newforms Admin
- SelectMultiple for a Google App Engine ListProperty using Django Forms
- Bulk Add/Replace from Command Line
- Starting a new Django Project with Versioning
- Compress a PDF with pdftk
- Ignore .pyc files in Subversion
- Backing up with BackupPC
- Quick image resizing with python

LINKS

options for the 'pdftk' binary) which should simply uncompress and recompress the data within the pdf to produce a smaller pdf — but not as small as the improvement you got.

Oscar Sandoval Torres | November 28, 2008 at 3:02 pm | Reply

This method not always work, in my case the resulted PDF is bigger.

attila82 | January 24, 2009 at 5:41 pm | Reply

Very good!
thanks

Pingback: Compressione di un pdf con minima perdita di qualità « Il blog di Andrea Lazzarotto

Paco | June 26, 2009 at 2:39 pm | Reply

I don't know what happends during the process but it helps me a lot
🙂 Great!
Thanks

caqui | July 21, 2009 at 2:40 am | Reply

it blanked the top edges of all pages…

Les | August 4, 2009 at 1:16 am | Reply

Thank you sooooo very very much for those instructions! I had this pdf file that was, believe it or not, 747 Mb in size. After using your instructions, I got myself a file that was ONLY 11 Mb in size without any loss in quality to the file! Thank you again!

ubundom | November 29, 2009 at 1:17 am | Reply

handy 6MB->227KB

Mitsos | December 1, 2009 at 9:12 am | Reply

Its one and a half years since your post…but still helping people 😊

Thanks.

Phlogi | December 12, 2009 at 3:46 pm | Reply

Thanks!

g___ | January 17, 2010 at 6:58 pm | Reply

7.2 MB -> 360 MB -> 14.3 MB ☹

iv_ru | February 28, 2010 at 7:17 pm | Reply

Hmm…

with this howto I got 4.9 MB from 1.4 MB in size! Something went wrong!

LaPeGa | July 23, 2010 at 5:59 pm | Reply

Install ghostscript and:

gs -sDEVICE=pdfwrite -dCompatibilityLevel=1.4 -

dPDFSETTINGS=/screen -dNOPAUSE -dQUIET -dBATCH -sOutputFile=output.pdf input.pdf

anemos | September 6, 2010 at 10:14 pm | Reply

Nice tip.
One thing though, I don't believe pdftk is required here. In fact, you are not even using it. The commands pdf2ps and ps2pdf are part of ghostscript.

Too bad it didn't help me, my 42 mb file went up to 7.9 gb and down to 1.4 gb ☹

constance | November 22, 2010 at 4:20 am | Reply

It doesn't work in my case, which generated a bigger file…

Riley | December 10, 2010 at 10:44 pm | Reply

that worked great! Best solution yet! Crushed my 27 mb down to 6.8mb

Pingback: hx4

Pablo Fonseca | January 14, 2011 at 11:23 pm | Reply

Great! Thanks

Santiago Restrepo | January 20, 2011 at 4:51 pm | Reply

It's just incredible. Has one PDF made of 800×600 photographs of a document = 75 Mb. After your treatment = 23.1 Mb.
I thought if I looped the process it were going to be like 10 Mb. It was not. It was = 23.1 Mb. So, no fractal allowed!

Thanks!

Simon Sparks | February 7, 2011 at 10:31 pm | Reply

Still a great tip! Thanks.
(90MB -> 15MB)

Sergiy | February 10, 2011 at 3:24 pm | Reply

THANK YOU VERY MUCH!
SOURCE was 645 MB ==> RESULT is 7,5 MB

belg4mit | March 7, 2011 at 11:38 pm | Reply

You can also use ImageMagick's convert or mogrify tools to downsample a PDF. e.g; convert foo.pdf -colors 2 -density 150×150 bar.pdf might turn a 5.5 Mb, 300dpi grayscale PDF into a 457kb 150dpi B&W PDF. However you get more readable results at the expense of larger output (1.4Mb) if converting to B&W with ps2pdf foo.pdf | pdf2ps – bar.pdf and mogrifying the results.

> **belg4mit** | March 7, 2011 at 11:41 pm | Reply
>
> Note that this is for image-based PDFs e.g; scanned pages

keyanm | April 15, 2011 at 11:23 pm | Reply

pdf2ps and ps2pdf are not a part of pdftk

orschiro | April 19, 2011 at 6:54 am | Reply

Interesting. It is working well.

Thanks for this tip.

carlos | April 26, 2011 at 10:36 am | Reply

From 220Mb pdf to 1,9 Gb ps to a 72Mb PDF!
Thank you!

Simon Sunnyboy / Paradize | June 10, 2011 at 7:53 pm | Reply

This worked well for me here. I had a 3,3MB PDF that contained a few scanned pages beside text. This trick reduced the size to 1,4MB instead!

Thanks for sharing, even if it does not work for every document, it is still worth a try!

FabioB | October 3, 2011 at 11:09 pm | Reply

gs -dNOPAUSE -sDEVICE=pdfwrite -r150 -sOUTPUTFILE=merged.pdf -dBATCH *.pdf

>>>(-r150)<<< set output file to 150dpi
-r100 set 100dpi etc etc…

my 2 cents

Markus | December 6, 2011 at 4:23 pm | Reply

Thanks, one PDF went from 258MB to 8MB. In case you are using Python 2.7, here's a little script that provides a very basic GUI. Creates two folders, one for the PS file, one for the compressed PDF. Hope the indentations display properly…

++++++++++++++++++++++++++++++++++++

```
import os, subprocess, Tkinter, tkFileDialog, tkSimpleDialog

from Tkinter import *

root = Tk()
root.title("PDF Compressor")

def compressPdf():
volumeDirectory = str(tkFileDialog.askdirectory(title="Select
Volume"))
pathName, directoryName = os.path.split(volumeDirectory)
directoryNameNew = directoryName + '_compressed'
directoryPathNew = os.path.join(pathName, directoryNameNew)
psDirectoryName = directoryName + '_ps'
psDirectoryPath = os.path.join(pathName, psDirectoryName)
if os.path.exists(directoryPathNew):
pass
else:
os.makedirs(directoryPathNew)
if os.path.exists(psDirectoryPath):
```

```
pass
else:
os.makedirs(psDirectoryPath)
pdfFiles = os.listdir(volumeDirectory)
for pdfFile in pdfFiles:
pdfFileName, extension = pdfFile.split(“.”)
psFileName = pdfFileName + ‘.ps’
pdfOldPath = os.path.join(pathName, directoryName, pdfFile)
psPath = os.path.join(psDirectoryPath, psFileName)
pdfNewPath = os.path.join(directoryPathNew, pdfFile)
subprocess.call(["pdf2ps", pdfOldPath, psPath])
subprocess.call(["ps2pdf", psPath, pdfNewPath])
print pdfNewPath
print ‘++++++++++++++++++’
print ‘Done!’

def close():
root.destroy()

menu = Menu(root)
root.config(menu=menu)
## File Menu
filemenu = Menu(menu)
menu.add_cascade(label=”File”, menu=filemenu)
filemenu.add_command(label=”Exit”, command=close)
## Process Menu
processmenu = Menu(menu)
menu.add_cascade(label=”Process Data”, menu=processmenu)
processmenu.add_command(label=”Compress PDFs”,
command=compressPdf)

mainloop()
```

atom | December 23, 2011 at 3:08 pm | Reply

Cool trick.

I just tried it with poor results – The output was 10x bigger than the input. But my ubuntu box has “pdf2ps” & “pdftops”. Using the latter instead of the former gave good results.

I have no idea why this works, but it’s a cool trick. Thanks!

Here’s another thing that shrinks a PDF where other tools fail… At least it worked with the file that lead me to this page 😊

gs -dNOPAUSE -dBATCH -dSAFER -sDEVICE=pdfwrite -dCompatibilityLevel=1.4 -sOutputFile=out-file.pdf in-file.pdf

YMMV.

This should work, but on some files it doesn’t… Which lead me to this page 😊

pdftk in-file.pdf output out-file.pdf compress

maiman | March 28, 2012 at 3:38 pm | Reply

almost 4 years from your post and still useful.

pdftops & ps2pdf: 20mb -> 133mb -> 1.5mb.

Alf | December 16, 2012 at 6:25 pm | Reply

simple and effective, thank you 🙂

I'm curious how often I can use this on one file….

> **jamstooks** | January 2, 2013 at 2:19 am | Reply
>
> Let me know what you find…

Pingback: Reducing PDF file size « Unforgettable Virtual Park

Boris | March 26, 2013 at 4:13 pm | Reply

Here's a bitcookie 8635 for you for your great tip.

LEAVE A REPLY

Email (required) (Address never made public)

Name (required)

Website

Notify me of follow-up comments via email.

Post Comment

Blog at WordPress.com. The Coraline Theme.